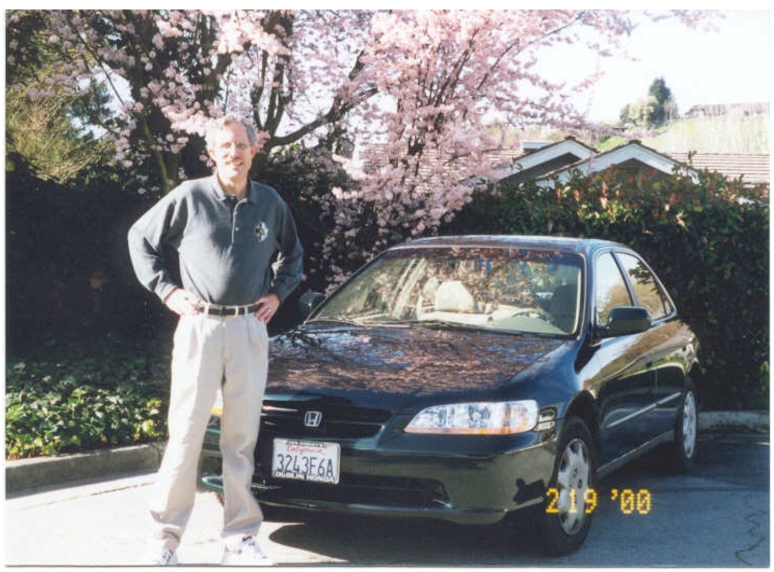
Are the Digits of Pi “Random”?

David H Bailey

Lawrence Berkeley National Laboratory



Acknowledgements

- Principal Collaborator:
  - Richard Crandall (Center for Advanced Computation, Reed College, USA)
- Others who have made important contributions:
  - Jonathan Borwein (Simon Fraser University, Canada)
  - Peter Borwein (Simon Fraser University, Canada)
  - David Broadhurst (Open University, United Kingdom)
  - Helaman Ferguson (Center for Computing Sciences, USA)
  - Simon Plouffe (Montreal, Canada)

Fascination with Digits of Pi

Newton (1670):

- "I am ashamed to tell you to how many figures I carried these computations, having no other business at the time."

Carl Sagan (1986):

- In his book "Contact," the lead scientist (played by Jodie Foster in the movie) looked for patterns in the digits of pi, alongside her research into extra-terrestrial intelligence.

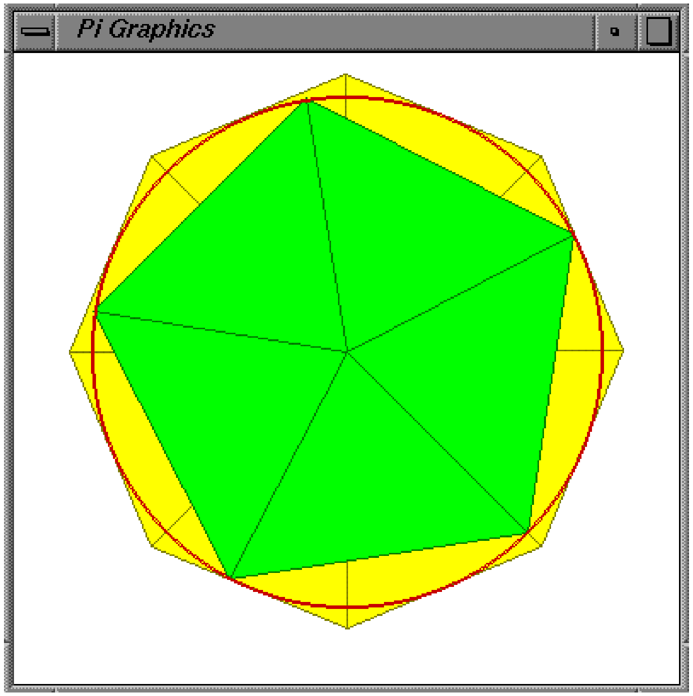
Carl E. Sagan

History of Pi Calculations

	Date	Correct Digits after Decimal Point
Babylonians	2000? BCE	0
Hebrews (1 Kings 7:23)	550? BCE	0
Archimedes	250? BCE	2
Liu Hui	263	5
Tsu Ch'ung Chi	480	7
Al-Kashi	1429	14
Romanus	1593	15
Van Ceulen	1615	35
Sharp	1699	71
Machin	1706	100
Strassnitzky and Dase	1844	200
Shanks	1874	527
Reitwiesner et al. (ENIAC)	1949	2,037
Shanks and Wrench	1961	100,265
Guilloud and Bouyer	1973	1,001,250
Kanada, Yoshino and Tamura	1982	16,777,206
DHB	1986	29,360,000
David and Gregory Chudnovsky	1989	480,000,000
Kanada and Tamura	1989	1,073,741,799
David and Gregory Chudnovsky	1991	2,260,000,000
Kanada and Takahashi	1997	51,539,600,000
Kanada and Takahashi	1999	206,158,430,000

Traditional Methods for Calculating Pi

- Archimedes (~250 BCE) computed pi to 2 decimal digits by computing areas of inscribed and circumscribed circles.



- Machin (1706) used the formula

$$\pi = 16 \tan^{-1}(1/5) - 4 \tan^{-1}(1/239)$$

where

$$\tan^{-1}(x) = x - x^3/3 + x^5/5 - x^7/7 + \ldots$$

Graphic: Peter Alfeld, Univ. of Utah

20th Century Algorithms for Pi

- 1976: Eugene Salamin and Richard Brent independently found a quadratically convergent algorithm for pi.
    - Each iteration approximately doubles the number of correct digits.
- 1985: Jonathan and Peter Borwein found a quartically convergent algorithm for pi.
    - Each iteration approximately quadruples the number of correct digits.

- The above two algorithms have been used in most of the recent record-breaking calculations of pi.

First 1000 Decimal Digits of Pi

3.
14159265358979323846264338327950288419716939937510
58209749445923078164062862089986280348253421170679
82148086513282306647093844609550582231725359408128
48111745028410270193852110555964462294895493038196
44288109756659334461284756482337867831652712019091
45648566923460348610454326648213393607260249141273
72458700660631558817488152092096282925409171536436
78925903600113305305488204665213841469519415116094
33057270365759591953092186117381932611793105118548
07446237996274956735188575272489122793818301194912
98336733624406566430860213949463952247371907021798
60943702770539217176293176752384674818467669405132
00056812714526356082778577134275778960917363717872
14684409012249534301465495853710507922796892589235
42019956112129021960864034418159813629774771309960
51870721134999999837297804995105973173281609631859
50244594553469083026425223082533446850352619311881
71010003137838752886587533208381420617177669147303
59825349042875546873115956286388235378759375195778
18577805321712268066130019278766111959092164201989

First 1000 Hexadecimal (Base 16) Digits of Pi

3.
243f6a8885a308d313198a2e03707344a4093822299f31d00
82efa98ec4e6c89452821e638d01377be5466cf34e90c6cc0
Ac29b7c97c50dd3f84d5b5b54709179216d5d98979fb1bd13
10ba698dfb5ac2ffd72dbd01adfb7b8e1afed6a267e96ba7c
9045f12c7f9924a19947b3916cf70801f2e2858efc1663692
0d871574e69a458fea3f4933d7e0d95748f728eb658718bcd
5882154aee7b54a41dc25a59b59c30d5392af26013c5d1b02
3286085f0ca417918b8db38ef8e79dcb0603a180e6c9e0e8b
B01e8a3ed71577c1bd314b2778af2fda55605c60e65525f3a
a55ab945748986263e8144055ca396a2aab10b6b4cc5c3411
41e8cea15486af7c72e993b3ee1411636fbc2a2ba9c55d741
831f6ce5c3e169b87931eafd6ba336c24cf5c7a3253812895
86773b8f48986b4bb9afc4bfe81b6628219361d809ccfb21a
991487cac605dec8032ef845d5de98575b1dc262302eb651b
8823893e81d396acc50f6d6ff383f442392e0b4482a484200
469c8f04a9e1f9b5e21c66842f6e96c9a670c9c61abd388f0
6a51a0d2d8542f68960fa728ab5133a36eef0b6c137a3be4b
a3bf0507efb2a98a1f1651d39af017666ca593e82430e888c

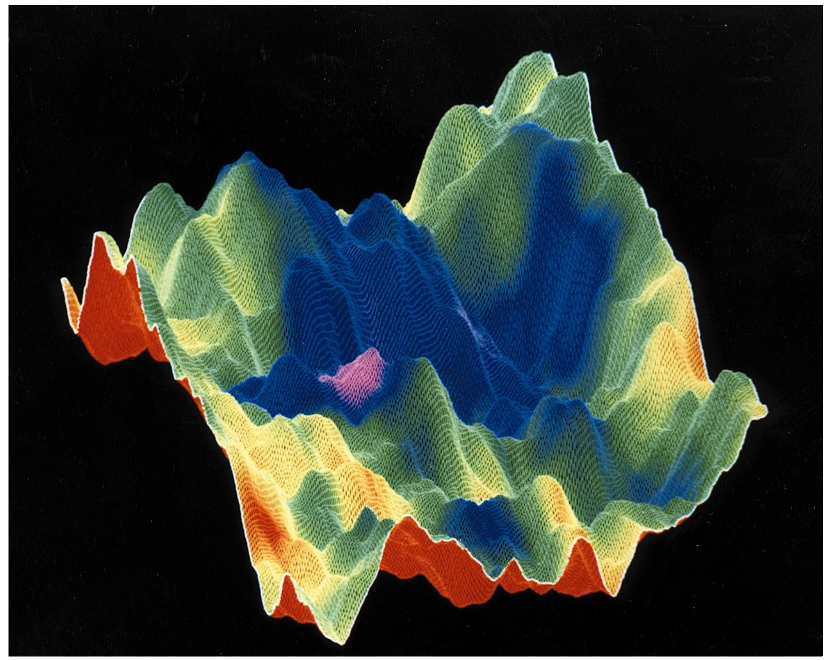
Is My Name in Pi?

The following website permits one to enter a name (or any other alphabetic or hexadecimal string), and see if it appears encoded in the first four billion binary digits of pi:

`http://pi.nersc.gov`

Centuries-Old Questions

- Are the digits of pi “random”?
- If so, why?



Graphic: Gregory and David Chudnovsky

Normal Numbers

- A number is normal base b if every string of m digits in the base-b expansion appears with frequency b^{-m}.
- Using measure theory, it is fairly easy to show that almost all real numbers are normal to any base.
- Widely believed to be normal base 10 (in fact normal base b for any b):
  - $\pi = 3.1415926535\ldots$
  - $e = 2.7182818284\ldots$
  - $\mathrm{Sqrt}(2) = 1.4142135623\ldots$
  - $\mathrm{Log}_e(2) = 0.6931471805\ldots$
  - Irrational roots of integer algebraic equations
  - Many others
  - But to date there have been NO proofs for any of these constants – not even a theoretical approach.

Some Applications of High Precision Arithmetic

- Climate modeling.
  - Requires 32 digit precision in some calculations.
- Vortex roll-up simulations.
  - Requires 64 digit precision in most calculations.
- Calculations of bound state spectra in Coulomb three-body systems.
  - Requires 80-100 digit precision in most calculations.

- Using the PSLQ algorithm to find previously unknown formulas in mathematics and physics.
  - Requires hundreds to many thousands of digit precision in most calculations.

LBNL's High-Precision Arithmetic Software Library

- Written entirely in C++ for portability and performance.
- Double-double (32 digits), quad-double, (64 digits) and arbitrary precision (>64 digits) versions are available.
- Special routines for extra-high precision (>1000 digits).
- Includes many common math functions: sqrt, cos, exp, log, etc.
- Includes several PSLQ and definite integral programs.
- C++ and Fortran-90 translation modules permit programmers to use this software with only minor changes to source code.

Authors: Brandon Thompson (UCB), Sherry Li (LBNL) Yozo Hida (UCB) and myself.

The PSLQ Integer Relation Algorithm

Let $(x_1, x_2, \ldots, x_n)$ be a vector of real numbers (computed to high precision). An integer relation algorithm finds integers (a_k) such that

$$a_1 x_1 + a_2 x_2 + a_3 x_3 + \ldots + a_n x_n = 0$$

- At the present time, the PSLQ algorithm of mathematician-sculptor Helaman Ferguson is the most efficient algorithm for integer relation detection.
- PSLQ was named one of 10 “algorithms of the century” by *Computing in Science and Engineering*.
- High precision arithmetic is required:
  - At least *d* x *n* digits, where *d* is the size (in digits) of the largest of the integers a_k.

One of Ferguson's Mathematical Sculptures

The PSLQ Algorithm

```
Initialize:  For j := 1 to n: for i := 1 to n: if i = j then set A_{i j} := 1 and B_{i j}
    := 1 else set A_{i j} := 0 and B_{i j} := 0; endfor; endfor.  For k := 1 to n: set
    s_k := sqrt {sum_{j=k}^n x_j^2}; endfor.  Set t := 1 / s_1.  For k := 1 to n: set y_k
    := t x_k; s_k := t s_k; endfor.
Initial H:  For j := 1 to n-1: for i := 1 to j-1: set H_{i j} := 0; endfor; set H_{j j}
    := s_{j+1}/s_j; for i := j+1 to n: set H_{i j} := - y_i y_j / (s_j s_{j+1}); endfor;
    endfor.
Reduce H:  For i := 2 to n: for j := i-1 to 1 step -1: set t := nint (H_{i j} / H_{j j});
    and y_j := y_j + t y_i; for k := 1 to j: set H_{i k} := H_{i k} - t H_{j k}; endfor;
    for k := 1 to n: set A_{i k} := A_{i k} - t A_{j k} and B_{k j} := B_{k j}
    + t B_{k i}; endfor; endfor; endfor.


Iterate:
Select m such that gamma^i |H_{i i}| is maximal when i = m.  Exchange the entries of y
    indexed m and m + 1, the corresponding rows of A and H, and the corresponding columns
    of B.
Remove corner on H diagonal:  If m <= n-2 then set t_0 := sqrt{H_{m m}^2 + H_{m,m+1}^2},
    t_1 := H_{m m} / t_0 and t_2 := H_{m,m+1} / t_0; for i := m to n: set t_3 := H_{i m},
    t_4 := H_{i, m+1}, H_{i m} := t_1 t_3 + t_2 t_4 and H_{i, m+1} := - t_2 t_3 + t_1
    t_4; endfor; endif.
Reduce H: For i := m+1 to n: for j := min(i-1, m+1) to 1 step -1: set t := nint (H_{i j}
    / H_{j j}) and y_j := y_j + t y_i; for k := 1 to j: set H_{i k} := H_{i k}
    - t H_{j k}; endfor; for k := 1 to n: set A_{i k} := A_{i k} - t A_{j k} and B_{k j}
    := B_{k j} + t B_{k i}; endfor; endfor; endfor.
Termination test:  If some y_i < epsilon, then a relation has been detected and is given
    in the corresponding column of B.
```

Identifying Algebraic Numbers Using PSLQ

Question: Is α *algebraic* (the root of an algebraic equation with integer coefficients) of degree *n* or less?

Solution: Compute the set of numbers

$(1, \alpha, \alpha^2, \alpha^3, \ldots, \alpha^n)$

to high precision, and then input these numbers to the PSLQ algorithm.

If a solution is found, the resulting integers are precisely the coefficients of the polynomial satisfied by α.

Application: Bifurcation Points in Chaos Theory

$B_3 = 3.54409035955\ldots$ is third bifurcation point of the "logistic iteration" of chaos theory:

$$x_{n+1} = r x_n (1 - x_n)$$

i.e., B_3 is the least value of r such that the iteration exhibits 8-way periodicity instead of 4-way periodicity.

A predecessor algorithm to PSLQ found that B_3 satisfies the polynomial

$$\begin{aligned} 0 = {} & 4913 + 2108t^2 - 604t^3 - 977t^4 + 8t^5 + 44t^6 \\ & + 392t^7 - 193t^8 - 40t^9 + 48t^{10} - 12t^{11} + t^{12} \end{aligned}$$

Example of the Many New Formulas Found by PSLQ

$$
\begin{aligned}
\tan^{-1}\left(\tfrac{2}{5}\right) = \frac{1}{67108864}\sum_{k=0}^{\infty}\frac{1}{2^{28k}}\Big(&\frac{67108864}{56k+1} - \frac{67108864}{56k+2} - \frac{33554432}{56k+4} \\
&- \frac{16777216}{56k+5} + \frac{29360128}{56k+7} + \frac{419304}{56k+9} - \frac{419304}{56k+10} - \frac{2097152}{56k+12} - \frac{1048576}{56k+13} \\
&- \frac{3670016}{56k+14} + \frac{262144}{56k+17} - \frac{262144}{56k+18} - \frac{131072}{56k+20} + \frac{163840}{56k+21} + \frac{16384}{56k+25} \\
&- \frac{16384}{56k+26} - \frac{8192}{56k+28} - \frac{4096}{56k+29} + \frac{1024}{56k+33} - \frac{1024}{56k+34} - \frac{1792}{56k+35} \\
&- \frac{512}{56k+36} - \frac{256}{56k+\_37} + \frac{64}{56k+41} + \frac{160}{56k+42} - \frac{32}{56k+44} - \frac{16}{56k+45} \\
&- \frac{10}{56k+49} - \frac{4}{56k+50} - \frac{2}{56k+52} - \frac{1}{56k+53}\Big)
\end{aligned}
$$

Some Large PSLQ Solutions

- Identification of B_4, the fourth bifurcation point of the logistic iteration.
  - Integer relation of size 121; 10,000 digit arithmetic.
- Identification of $S(20)$, where $S(k) = \Sigma_{n>0} 1/(n^k C(2n,n))$.
  - Integer relation of size 118; 5,000 digit arithmetic.
- Identification of Euler-zeta sums.
  - Hundreds of integer relation problems, each of size 145 and requiring 5,000 digit arithmetic.
  - Done on IBM SP parallel computer system.
- Discovery of a relation involving a root of Lehmer's polynomial.
  - Integer relation of size 125; 50,000 digit arithmetic.
  - Done on the NERSC Cray T3E parallel computer -- 48 hours on 32 CPUs.

The BBP Formula for Pi

In 1996, PSLQ was used to discover a previously unknown formula for pi:

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right)$$

This formula permits one to directly calculate the *n*-th binary or hexadecimal (base 16) digit of pi, without needing to calculate any of the first *n* -1 digits.

So simple! Why wasn't it found hundreds of years ago?

Proof of the BBP Formula

$$\int_0^{1/\sqrt{2}} \frac{x^{j-1}}{1-x^8} = \int_0^{1/\sqrt{2}} \sum_{k=0}^{\infty} x^{8k+j-1} dx = \frac{1}{2^{j/2}} \sum_{k=0}^{\infty} \frac{1}{16^k (8k+j)}$$

Thus we can write

$$\sum_{k=0}^{\infty} \frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right)$$

$$= \int_0^{1/\sqrt{2}} \frac{4\sqrt{2} - 8x^3 - 4\sqrt{2}x^4 - 8x^5}{1-x^8} dx$$

$$= \int_0^1 \frac{16(4 - 2y^3 - y^4 - y^5)}{16 - y^8} dy$$

$$= \pi$$

A straightforward exercise in calculus!

Using the BBP Formula to Find Individual Hex Digits of Pi

Let S_1 be the first of the four sums in the formula for π, and let $\{\}$ denote fractional part. Then the hex expansion of S_1 starting at position $d+1$ is

$$\left\{16^d S_1\right\} = \left\{\sum_{k=0}^{\infty} \frac{16^{d-k}}{8k+1}\right\} = \left\{\sum_{k=0}^{d} \frac{16^{d-k} \bmod (8k+1)}{8k+1} + \sum_{k=d+1}^{\infty} \frac{16^{d-k}}{8k+1}\right\}$$

Note that the numerator $16^{d-k} \bmod (8k+1)$ can be computed very rapidly by means of the binary algorithm for exponentiation, where multiplications are performed modulo $8k+1$.

By repeating this computation for S_1, S_2, S_3 and S_4, and calculating $4S_1$ - $2S_2$ - S_3 - S_4, one obtains a fraction that has several hexadecimal digits of π, beginning at position $d+1$.

This algorithm requires very little memory, and only 64-bit or 128-bit floating-point arithmetic.

Calculations Using the BBP Formula

Position	Hex Digits of Pi Starting at Position	
10^6	`26C65E52CB4593`	
10^7	`17AF5863EFED8D`	
10^8	`ECB840E21926EC`	
10^9	`85895585A0428B`	
10^{10}	`921C73C6838FB2`	
10^{11}	`9C381872D27596`	
1.25×10^{12}	`07E45733CC790B`	[1]
2.5×10^{14}	`E6216B069CB6C1`	[2]

[1] Babrice Bellard, France, 1999

[2] Colin Percival, Canada, 2000

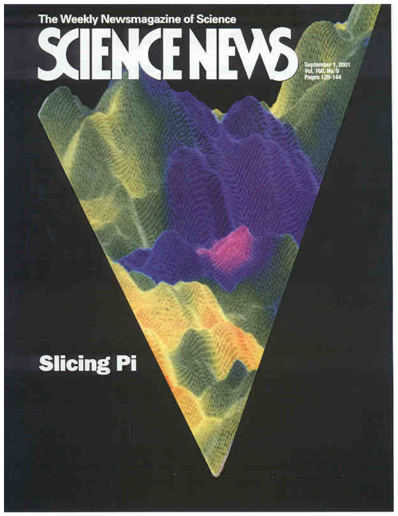
New Result (Feb. 2001)

In 2001 Richard Crandall and I demonstrated that:

The question of whether certain math constants [including π and log(2)] are normal reduces to a plausible hypothesis from the field of chaotic dynamics.

This result relies crucially on the BBP formula for pi and some other similar formulas discovered using PSLQ.

See *Science News*, Sept. 1, 2001 for some additional background of this discovery.



Hypothesis A and Its Consequence

Hypothesis A. Let $p(k)$ and $q(k)$ be integer polynomials, where $\deg(p) < \deg(q)$, and where q has no zeroes for $k > 0$. Let $r_n = p(n)/q(n)$, and let $b > 1$ be an integer. Set $x_0 = 0$. Then the sequence

$$x_n = \{b x_{n-1} + r_n\}$$

either has a finite attractor or is equidistributed in (0,1).

Theorem. Assuming Hypothesis A, then any constant α given by a formula of the form

$$\alpha = \sum_{k=0}^{\infty} \frac{p(k)}{b^k q(k)}$$

(where p and q are integer polynomials as above) is either normal base b or rational.

Example

Consider the sequence (x_n) given by

$$x_n = \{2x_{n-1} + 1/n\}$$

where { } denotes fractional part as before. Succesive values of (x_n) appear to dance about randomly in the interval (0,1):

0.0000, 0.5000, 0.3333, 0.9167, 0.0333, 0.2333, 0.6095, 0.3440,
0.7992, 0.6984, 0.4877, 0.0588, 0.1945, 0.4605, 0.9876, 0.0378,
0.1344, 0.3243, 0.7012, 0.4524, 0.9524, 0.9502, 0.9439, 0.9295,
0.8990, 0.8364, 0.7098, 0.4553, 0.9451, 0.9236, 0.8794, 0.7901,...

Crandall and I have shown that if it can be proven that (x_n) is truly equidistributed in the interval (0,1), then it would follow that log(2) is normal base 2.

A Sequence for Pi

The sequence corresponding to π is

$$x_n = \{16x_{n-1} + \frac{120n^2 - 89n + 16}{512n^4 - 1024n^3 + 712n^2 - 206n + 21}\}$$

As before, if it could be proved that the sequence (x_n) is truly equidistributed in (0,1), this would suffice to establish that π is normal base 16 (and thus also base 2).

Curious fact: The above sequence, if converted to hexadecimal digits by multiplying each x_n by 16 and taking integer parts, appears to precisely generate the entire hex expansion of π, without any errors. We do not fully understand why.

Latest Result (Mar. 2002)

Crandall and I have now shown (unconditionally) that an infinite class of mathematical constants, including

$$\alpha = \sum_{k=1}^{\infty} \frac{1}{3^k 2^{3^k}}$$

= 0.041883680831502985… (decimal)

= 0.0AB8E38F684BD… (hexadecimal)

is normal. In particular, this α is normal base 2.

Note: The particular α above was proven to be normal base 2 by Stoneham in 1973, but our results cover a much larger class.

Other Properties of the New Normal Constants

- Transcendental: Like pi, e and log(2), they are not roots of any polynomials with integer coefficients.
- Individual digit calculation property: Like pi and log(2), arbitrary individual digits can be rapidly calculated.

Example calculation:

The googol-th (i.e., the 10^{100}-th) binary digit of α is 0.

Summary

- Are the digits of pi "random"? -- i.e., is pi normal?
  - Still an open question, although we have a partial result for hex and binary (but not decimal) digits of pi.
  - Normality has been proven for certain other numbers.
- Why?
  - Because the hex and binary digits of pi, and the digits of certain other numbers, are approximately generated by chaotic pseudorandom generators.
  - Indeed, the question of normality of certain math constants reduces to a question regarding the properties of these pseudorandom generators.

Are There Any "Practical" Applications of These Results?

Not yet. But some possibilities include:

- Pseudorandom number generation for certain physics applications.
- Cryptography (secure communication).

Recall mathematician G. H. Hardy's 1943 comment:

I have never done anything 'useful'.

- In contrast, today's secure website technology is based on mathematical number theory, including many of Hardy's results.

For Further Information

- Several papers on these topics, both “popular” and technical, are available from the website

  `http://www.nersc.gov/~dhbailey/dhbpapers`

- LBNL’s high-precision arithmetic software (with C and Fortran modules), together with implementations of PSLQ and other related programs, are available at

  `http://www.nersc.gov/~dhbailey/mpdist/mpdist.html`

- The pi digit search facility is available at
  `http://pi.nersc.gov`